Chapter 8

First-Order Logic Modulo Theories

In this section I generalize first-order superposition with equality, Chapter 5, to theories, in particular superposition modulo LA.

8.1 Introduction

First-order logic equational clauses are extended with constraints over some theory $\mathcal{T}$, see Definition 3.16.1. For example, the clause

$$x > 5 \wedge y = 3x - 1 \parallel R(x, y, g(z)) \vee g(z) \approx z$$

is a combination the LRA with first-order equational logic. For convenience, I write non-equational atoms like $R(x, y, g(z))$ in a predicative way, but of course, they are an abbreviation for the respective equations, i.e., $f_R(x, y, g(z)) \approx$ true in case of $R(x, y, g(z))$. The clause abbreviates the formula

$$\forall x, y, z.((x > 5 \wedge y = 3x - 1 + a) \rightarrow R(x, y, g(z)) \vee g(z) \approx z)$$

where a is a constant of sort LA as well as the variables x, y, and z may be be of some sort outside LA, e.g., some sort S. So the fragment is a strict generalization of the fragment considered by the Nelson-Oppen theory combination, Section 7.1, because there are universally quantifier variables. Furthermore, it becomes obvious that a many-sorted setting is needed for this logic in order to distinguish theory variables from first-order variables.

The above writing $\lambda \parallel C$ emphasises the separation of the first-order part C of the clause and the theory part λ of the clause. The idea of the calculus is to process C using superposition and to check satisfiability of ϕ by a separate procedure for the theory. It is therefore called the *hierarchic* superpositon claculus because the reasoning is driven by the first-order part. In case of LA by the decision procedures discussed in Section 6.2, so for the purpose of this chapter and the case of LA I always assume that λ is a conjunction of LA atoms.

If the LA constraint $x > 5 \wedge y = 3x - 1$ is not satisfiable, considering x, y to be existentially quantified, then the clause $x > 5 \wedge y = 3x - 1 \parallel$

$R(x, y, g(z)) \vee g(z) \approx z$ is a tautology. The idea of the hiearchic superposition calculus is to derive the empty constraint clause $\phi \parallel \bot$ by superposition and then check satisfiability of ϕ to eventually justify the refutation. Basically, that's the idea of the calculus and is a very natural way of thinking about reasoning of a combination. However, the resulting language is very expressive resulting in additional needed assumptions for a complete calculus.

The LRA clauses over a simple unary first-order predicate N actually encode the natural numbers.

$$\begin{aligned} &\parallel N(0)\\ x < 0 &\parallel \neg N(x)\\ y = x + 1 &\parallel \neg N(x) \vee N(y)\\ 0 < x \wedge x < 1 &\parallel \neg N(x)\\ x > 1 \wedge y = x - 1 &\parallel \neg N(x) \vee N(y) \end{aligned}$$

The first clause says 0 is in N. The second that no negative rational number is in N. The third expresses that if x is in N, so is $x + 1$. The fourth excludes all rationals between 0 and 1 from N. Finally, the fifth clause together with the fourth clause excludes any rational number strictly larger than 1 that is not a natural number from N.

Note that both first-order logic as well as the first-order LRA theory are not expressive enough to encode the nataral numbers. The combination is. So the hierarchic superposition calculus can no longer be a semi-decision procedure for unsatsifiability, in general. Unsatisfiability of a hierarchic clause set is undecidable, in general.

Next the above clauses are extended by four further clauses with an additional LRA constant a and an additional free predicate P.

$$\begin{aligned} x = a &\parallel N(x)\\ &\parallel P(0)\\ y = x + 1 &\parallel \neg P(x) \vee P(y)\\ x = a &\parallel \neg P(x) \vee \neg N(x) \end{aligned}$$

The first clause expresses a is a natural number. But then from the next three clauses $n = a \parallel \bot$ is derivable, i.e., $n \neq a$, by superposition for any natural number n. There is no overall refutation since any finite conjunction of atoms $a \neq 1 \wedge a \neq 2 \wedge \ldots$ is satisfiable. Therefore, by adding a single constant of the arithmetic sort, i.e., an additional existentially quantified variable, the logic is not compact anymore. Note that just adding the clause $x = a \parallel \neg N(x)$ imemdiately yields a refutation with the clause $x = a \parallel N(x)$.

An alternative would be to move the constant a to the first-order side.

$$\begin{aligned} &\parallel N(a)\\ &\parallel P(0)\\ y = x + 1 &\parallel \neg P(x) \vee P(y)\\ &\parallel \neg P(a) \vee \neg N(a) \end{aligned}$$

But then after any inference involving $P(a)$, the constant a again moves into the arithmetic constraint. The way out is to forbid any unifier that substitutes a constant or function term for a variable of the arithmetic sort. In this case the only inference involving a results in the clause $\parallel \neg P(a)$ and there is no inference with this clause. Again there is no refutation. The calculus is not complete. The way out is to require *sufficient definedness* of first-order terms ranging into the arithmetic sort. Each such ground term must be reducible to a term only containing symbols from arithmetic. For the above example, the clause

$$\parallel a \approx 42$$

sufficiently defines a. Adding this clause to the above clauses yields a refutation by deriving the clauses

$$\begin{aligned} &\parallel \neg P(x) \lor \neg N(42)\\ x = 0 &\parallel \neg P(x) \lor \neg N(x)\\ x = 0 &\parallel \neg N(x)\\ x = 0 &\parallel \bot\\ &\bot \end{aligned}$$

where the first clause is a result of two Superposition Right applications on $\parallel \neg P(a) \lor \neg N(a)$ with $x = 42 \parallel a \approx x$. The second of 42 Superposition Left inferences with $y = x+1 \parallel \neg P(x) \lor P(y)$, and finally the resulting clause is refuted with the clauses $x = 0 \parallel P(x)$ and $x = 0 \parallel N(x)$.

Finally, consider the formula

$$\forall x, y\,.\,(x \approx y \lor g(x) \leq g(y) - 1 \lor g(x) \geq g(y) + 1)$$

over free first-order logic modulo linear arithmetic where x, y are variables of the linear arithmetic sort and g is a free first-order function. By first-order logic semantics the function g is total. So the formula is true for some interpretation of g if for any two different values x, y out of the LA domain, the distance between $g(x)$ and $g(y)$ is at least one. So the LA domain can be ordered by $g(n_1) < g(n_2) < g(n_3) < \ldots$ where the distance between subsequent $g(n_i)$ is at least one. Models of the formula induce an enumeration function for the LA domain. Hence, the formula is satisfiable over the rationals or integers but not over the reals. Again, neither the first-order theory over linear arithmetic nor free first-order logic can separate models over the reals from models over the rationals.